Détection et apprentissage automatique d'objets pour la modélisation de milieux intérieurs

Marion Decrouez, Romain Dupont, François Gaspard, Frédéric Devernay, James L. Crowley

Détection et apprentissage automatique d'objets pour la modélisation de milieux intérieurs

M. Decrouez[1,2] R. Dupont[1] F. Gaspard[1] F. Devernay[2] J.L. Crowley[2]

[1] CEA, LIST, Laboratoire Vision et Ingénierie des Contenus
[2] INRIA Grenoble - Rhône-Alpes

[1] Point Courrier 94, Gif-sur-Yvette, F-91191 France
[2] Avenue de l'Europe, 38330 Montbonnot-Saint-Martin
marion.decrouez@cea.fr

Résumé

Nous présentons dans cet article une nouvelle méthode pour la modélisation des objets et de la scène dans un environnement intérieur inconnu. Les milieux intérieurs sont composés d'une quantité d'objets susceptibles d'être déplacés. Nous souhaitons exploiter les multiples passages d'une caméra dans un même lieu et tirer parti de ces déplacements pour modéliser d'une part la structure statique de la scène et d'autre part les objets le constituant. Nous proposons une association de méthodes de SLAM métrique et de reconnaissance de lieu pour détecter et apprendre les objets de façon automatique et enrichir la connaissance de la scène.

Mots Clef

SLAM par vision, sac de mots, reconnaissance de lieu, détection d'objets.

Abstract

This paper presents a new solution for modeling the scene and the objects in unknown environments. Many objects in indoor environments are likely to be moved. We want to make the most of several observations of a camera in the same scene to represent the different places and objects. We propose to combine methods of metrical localization and place recognition to detect and model objects and extend the scene model.

Keywords

Visual SLAM, bag of words, place recognition, objects detection.

1 Introduction

Aujourd'hui, le traitement d'une séquence video prise par une caméra unique permet de modéliser une scène *a priori* inconnue en temps réel. Les algorithmes de SLAM (Simultaneous Localization And Mapping) métrique calculent la trajectoire de la caméra tout en reconstruisant une carte éparse des primitives visuelles de l'environnement. Ces approches sont la base d'applications de réalité augmentée [10], [9]. Les travaux actuels permettent d'incruster de façon réaliste des objets virtuels dans la séquence d'images et intéressent de nombreuses industries (jeux vidéos). L'approfondissement de ces systèmes permettra à l'avenir des applications industrielles précises comme l'aide à la maintenance ou à l'assemblage. Les environnements intérieurs sont constitués d'une multitude d'objets susceptibles d'être déplacés. La modélisation de milieux dynamiques comme les rayons d'un supermarché ou les ateliers d'une usine est perturbée par ces mouvements. Au lieu d'essayer de les filtrer, nous souhaitons en tirer parti pour détecter et apprendre de manière automatique les objets de l'environnement. L'approche présentée dans cet article propose de définir explicitement la scène comme une structure statique et un ensemble d'objets dynamiques. Sans autre information a priori, un objet est défini comme un ensemble de primitives visuelles ayant eu le même déplacement par rapport à la structure statique entre deux passages (approche illustrée figure 1). Nous exploitons les multiples passages de la caméra dans un même environnement pour tirer le maximum d'informations sur la scène et son évolution au fil du temps.

Le système présenté dans cet article permet de reconstruire un environnement en 3D et de s'y localiser, de reconnaître un lieu déjà visité et de modéliser automatiquement de nouveaux objets. La section 2 donne une vue d'ensemble des approches de SLAM vues dans la littérature. Nous présentons dans la section 3 l'algorithme de localisation et de reconstruction 3D. La section 4 décrit la détection automatique des objets. Enfin, nos résultats et perspectives sont présentés section 5.

2 Travaux antérieurs

Les approches répondant au problème du SLAM fondées uniquement sur la vision (Vision-SLAM) peuvent être classées en deux catégories : les approches métriques et les ap-

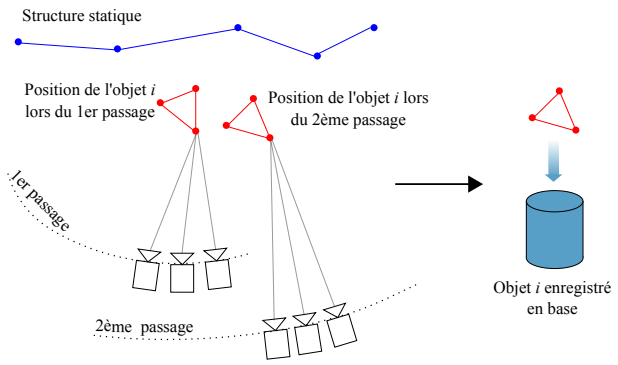


FIGURE 1 – Détection automatique d'objets. La comparaison des reconstructions 3D de différents passages dans un même lieu permet de détecter des objets déplacés.

proches topologiques. Les algorithmes de SLAM métrique permettent de reconstruire un environnement inconnu en 3D et de s'y localiser précisément. Les méthodes de SLAM topologiques et de reconnaissance de lieu reposent sur une représentation discrète de l'environnement modélisé sous forme d'un graphe. Les nœuds du graphe sont des lieux distincts et les arêtes représentent les relations entre lieux (positionnement relatif, adjacence temporelle).
Différentes solutions ont été proposées pour l'estimation des paramètres d'une scène 3D. Les méthodes fondées sur l'optimisation des paramètres par ajustement de faisceaux ont des résultats remarquables. L'ajustement de faisceaux global, permettant d'optimiser l'ensemble des paramètres de la scène, est la méthode de reconstruction 3D la plus précise mais elle ne permet pas une éxécution en temps réel. Mouragnon et al. [10] proposent d'utiliser une méthode d'ajustement de faisceaux local (seuls les paramètres des dernières caméras sont optimisés) et parviennent à reconstruire l'environnement en temps réel. Cette approche présente encore certaines limitations. On observe une dérive cumulative dans l'estimation de la pose de la caméra et une dérive du facteur d'échelle : les reconstructions sont réalisées à un facteur d'échelle théoriquement constant sur la séquence entière, mais on observe une dérive de ce facteur le long de la trajectoire. Il est alors difficile de repérer que la caméra repasse par un même endroit et de détecter les boucles dans la trajectoire. De même, ces algorithmes ne permettent pas de se relocaliser dans la carte après une perte de la position de la caméra.
Motivées par la détection de boucle, de nouvelles approches de SLAM topologique ou SLAM basé sur l'apparence ont été proposées dans la littérature. Ces méthodes considèrent le problème du SLAM comme un problème de reconnaissance d'images : deux images similaires proviennent probablement du même endroit. De nombreux systèmes de localisation reposent sur une représentation de l'image en sac de mots visuels. Ces méthodes inspirées des techniques de recherche d'information présentent l'image comme un ensemble de primitives visuelles, les mots, définis dans un dictionnaire ou vocabulaire [14], [11]. Les travaux de Cummins et Newman [6] définissent un formalisme probabiliste reposant sur l'approche en sac de mots visuels. L'environnement est un ensemble de lieux discrets, dont l'apparence est modélisée par une distribution sur les mots du dictionnaire. L'algorithme offre une robustesse remarquable à l'aliasing perceptuel grâce à la prise en compte des probabilités de co-occurrences des mots visuels (calculées hors-ligne) dans l'estimation de la vraisemblance de l'observation. Néanmoins le système n'autorise pas des traitements en temps réel. De plus, il a prouvé son efficacité sur des images provenant de caméras panoramiques et nous souhaitons utiliser des optiques de champ moyen.
De plus en plus de travaux combinent les deux approches pour gérer des trajectoires plus longues tout en maintenant une carte des points 3D nécessaire aux applications de réalité augmentée. Les auteurs de [5] propose un système gérant simultanément plusieurs cartes 3D et mettant en oeuvre la relocalisation de l'image courante. Ils recherchent parmi les images précédentes l'image la plus proche en utilisant une description globale de l'image. De nombreux travaux s'attachent aujourd'hui à extraire davantage d'informations du flux video pour mieux comprendre la scène, reconnaître un lieu déjà visité ou détecter un objet préalablement appris. Ils visent souvent à améliorer les résultats du SLAM. Castle et al. [4] construisent une base de données d'objets hors ligne pour les reconnaître et les utiliser comme points de repère robustes lors de la localisation. D'autres approches proposent de modéliser et d'apprendre les objets en ligne. Reitmayr [12] propose d'enrichir une base de données mais la méthode nécessite l'intervention de l'utilisateur qui doit segmenter manuellement les objets lors d'un premier passage. Angeli et al. [1] suggèrent de regrouper les primitives observées suivant leur similarité et leur proximité en *clusters* 3D. Ces *clusters* sont souvent associés à des objets réels de la scène mais ils ne permettent pas de définir un objet de manière robuste. Enfin, Kim et al [8] présentent une méthode de suivi de multiples objets en temps réel dans un environnement inconnu. Les objets sont préalablement appris et l'utilisateur peut enrichir la base de données en sélectionnant une région d'intérêt dans l'image. L'apprentissage des objets est donc manuel ou semi-automatique. A notre connaissance, aucune méthode ne permet d'enrichir une base de données d'objets en considérant de multiples passages dans le même environnement.
Notre méthode est résumée figure 2. A chaque nouvelle image clef, le nuage de points 3D est mis à jour et on recherche dans l'ensemble des images précédentes l'image la plus proche pour comparer les reconstructions et détecter des objets déplacés. Les sections suivantes présentent notre algorithme.

3 Modélisation de l'environnement

Nous présentons dans cette section notre algorithme pour la reconstruction 3D, la reconnaissance de lieu basée sur

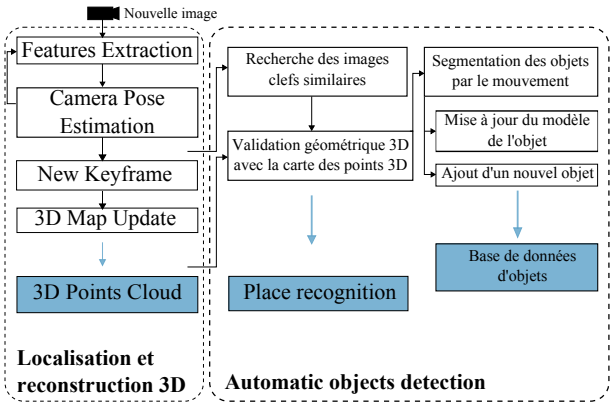


FIGURE 2 – Méthode proposée : Reconstruction 3D et apprentissage automatique des objets déplacés

l'apparence et la validation géométrique 3D unifiant ces deux méthodes.

3.1 Reconstruction 3D de la scène

Avec la méthode décrite dans [10], le traitement de la séquence video permet de construire une carte éparse des points 3D en temps réel. L'algorithme détecte les points de Harris-Stephen de l'image courante et extrait les descripteurs SURF [3]. Ces points sont appariés avec la projection des points 3D observés dans l'image précédente pour estimer la pose de la caméra puis leur coordonnées 3D par triangulation. Certaines images clefs sont sélectionnées pour mettre à jour la carte et optimiser le résultat par ajustement de faisceaux local, ce qui permet un traitement en temps réel. Cette méthode présente cependant des dérives dans l'estimation de la pose de la caméra et du facteur d'échelle qui perturbent la détection de boucles dans la trajectoire. Nous utilisons donc un algorithme de reconnaissance de lieu reposant sur l'apparence globale de l'image pour repérer un lieu déjà visité (section 3.2). En analysant les multiples passages dans un même endroit nous sommes capables d'identifier des points 3D ayant bougé (section 3.3), puis de détecter les objets déplacés (section 4).

3.2 Reconnaissance de lieu

Notre algorithme de reconnaissance de lieu repose sur la représentation de l'image en sac de mots visuels [11]. Cette approche permet de trouver des images similaires à une image requête au sein d'une grande base de données de façon rapide. Les descripteurs locaux de l'image sont quantifiés par des mots visuels d'un dictionnaire préalablement appris. Le dictionnaire de mots visuels est appris hors ligne en agglomérant les descripteurs SURF extraits d'un ensemble d'images suivant la méthode des k-means. Nous utilisons 3000 images aléatoires téléchargées sur Flickr car nous souhaitons modéliser tout type de scène. Les images clefs sélectionnées par l'algorithme de SLAM métrique sont progressivement enregistrées dans une base de données. Elles sont modélisées par un histogramme des fréquences d'apparition des mots visuels. Cette opération n'est pas coûteuse : nous recherchons les mots visuels présents dans l'image clef et nous modifions le fichier inverse répertoriant la liste des images contenant chaque mot visuel. L'utilisation d'un fichier inverse permet de calculer efficacement la similarité entre images. Pour chaque mot trouvé dans l'image courante, le score de similarité des images renvoyées par le fichier inverse est mis à jour en additionnant un terme inspiré de la méthode de pondération TF-IDF (Term Frequency - Inverse Document Frequency) :

$$tf - idf = p_{w_{L_i}} \log\left(\frac{1}{p_w}\right) \quad (1)$$

$p_{w_{L_i}}$ est la fréquence du mot w dans l'image. p_w est la probabilité d'occurrence du mot w, calculée lors de la construction du dictionnaire. $\frac{1}{p_w}$ est une mesure de l'importance du mot visuel w. Les mots apparaissant peu sont considérés comme plus discriminants et ont un poids plus important.

Les méthodes de SLAM topologique ne se contentent pas de retrouver des images similaires dans une base de donnée déterminée. Elles sont confrontés à la difficulté de détecter que l'image courante provient d'un nouveau lieu. Les auteurs de [2] définissent pour cela la notion d'image virtuelle qui représente une moyenne des lieux déjà visité. Si l'image courante ressemble plus à l'image virtuelle qu'aux autres images en base, un nouveau lieu est détecté et modélisé. Néanmoins cette méthode ne permet pas de rejeter toutes les fausses correspondances. La vérification de la cohérence géométrique améliore considérablement les performances [6] en rendant la reconnaissance robuste à l'aliasing perceptuel.

3.3 Unification des deux approches pour une validation géométrique 3D/2D

La reconnaissance de lieu nécessite une étape de vérification. En effet, cette méthode renvoie l'image enregistrée en base qui se rapproche le plus de l'image courante. Cependant, deux lieux distincts peuvent avoir la même apparence et on observe souvent des erreurs dues à ce phénomène appelé *aliasing perceptuel*. Pour améliorer la robustesse de la reconnaissance de lieu, on introduit plusieurs validations géométriques présentées ci-dessous.

Validation géométrique 2D/2D. En règle générale, les méthodes de SLAM topologique vérifient la cohérence géométrique 2D des deux images mises en correspondance. Certaines approches examinent la position relative des descripteurs dans les images. D'autres confirment la reconnaissance de lieu si une homographie ou une matrice fondamentale peut être calculée à partir des correspondances de points entre les images. Nous souhaitons utiliser la reconstruction des points 3D pour valider rigoureusement les hypothèses.

Validation géométrique 3D/2D. Nous validons les hypothèses de reconnaissance de lieu avec une vérification géométrique 3D qui unifie les approches de SLAM topologique et métrique. Les primitives extraites dans l'image

de la base de données sont appariées avec la projection des points 3D vus dans l'image courante et la pose relative de l'image de la base de données est calculée (figure 3). Le nombre de points vérifiant la contrainte géométrique permet alors de retenir l'hypothèse. Nous rejetons ainsi toutes les erreurs dues à l'*aliasing* perceptuel. Cette méthode permet surtout de mettre en évidence la structure statique de la scène d'une part et un ensemble de points incohérents d'autre part. Considérant que la scène consiste en une structure statique et un ensemble d'objets rigides en mouvement les uns par rapport aux autres, nous souhaitons regrouper les points de mouvement cohérent pour définir des objets. Ce problème est commmunément désigné par l'expression « segmentation par le mouvement ». Nous présentons des solutions lues dans la littérature section 4.1 puis notre méthode de détection d'objets section 4.2.

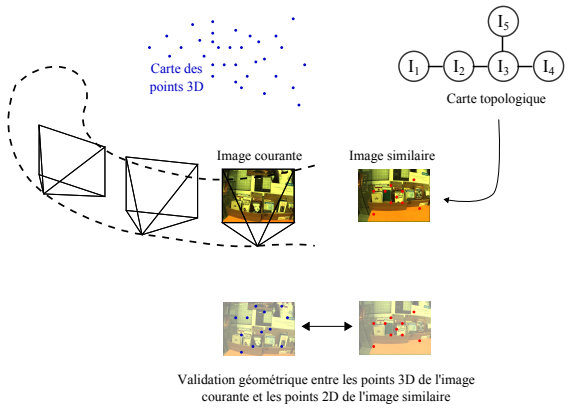


FIGURE 3 – Validation géométrique : les points d'intérêt de l'image similaire sont appariés avec les points 3D vus dans l'image courante et la pose relative de l'image similaire est calculée.

4 Détection automatique des objets

La comparaison de deux images provenant d'un même lieu et prises à des moments différents met en évidence des points 3D incohérents avec la structure statique de la scène. Nous inférons la présence d'objets ayant bougé en regroupant les points par mouvement similaire. Etant données les correspondances de points entre deux vues, il faut établir les différents mouvements de la scène : le mouvement de la caméra et le mouvement des objets. Pour simplifier notre problème nous considérons uniquement les objets plans. Dans cette section, nous faisons en premier lieu un bref état de l'art sur les méthodes de détection de multiples modèles entre deux vues et nous exposons ensuite notre approche.

4.1 Détection de multiples modèles entre deux vues

Le problème de l'estimation d'un modèle géométrique à partir d'un ensemble de données bruitées est très courant en vision par ordinateur. Il est généralement résolu avec l'utilisation de l'algorithme RANSAC. Notre problème est plus compliqué : les aberrations dans les données proviennent des erreurs d'appariemment et du bruit de mesure mais aussi du fait que plusieurs structures sont présentes dans la scène. D'autre part, plusieurs types de modèles peuvent coexister : le mouvement d'une surface plane est modélisé par une homographie, celui d'un objet 3D est modélisé par une matrice fondamentale. Schindler et al. [13] proposent d'estimer les différents mouvements d'une scène en se basant sur le critère de sélection de modèle GRIC décrit dans [16]. L'algorithme est coûteux et on observe des erreurs dans la sélection qui privilégie souvent le modèle le plus général. Nous simplifions notre problème en ne recherchant pour l'instant que les objets plans. Zuliani et al. [17] proposent l'algorithme MultiRANSAC pour détecter des homographies. La méthode présente des résultats satisfaisants mais l'utilisateur doit préciser le nombre de modèles. Toldo et Fusiello [15] proposent une méthode simple et rapide pour estimer les différentes instances d'un modèle à partir des correspondances entre deux vues. Un grand nombre d'homographies est généré à partir d'échantillons de correspondances tirées aléatoirement. Des ensembles de points appartenant au même modèle sont fusionnés suivant un algorithme appelé J-Linkage. Notre méthode, présentée en section 4.2, s'inspire de cet algorithme de fusion. Cependant nous utilisons une methode itérative pour limiter le nombre d'hypothèses générées avant la fusion. De plus nous tirons profit de la reconstruction 3D pour détecter uniquement les objets ayant bougé.

4.2 Identification des différents mouvements dans la scène 3D

Les cartes de points 3D reconstruites sont éparses. Si nous pouvons détecter un ensemble de points incohérents avec la structure statique de la scène, leur nombre de points est souvent insuffisant pour segmenter de manière robuste les objets. Pour pallier ce problème nous recherchons les homographies entre les deux vues avec un plus grand nombre de correspondances et nous retenons les ensembles contenant des points associés à des points 3D incohérents. Notre méthode est illustrée figure 4. Le calcul de pose relative entre deux images similaires sépare les points de la structure statique en bleu des points incohérents représentés par des étoiles rouges (figure 4 (a)). Nous recalculons un plus grand nombre de points dans les deux images et nous les apparions. Les surfaces planes et les objets plans de la scène sont détectés (figure 4 (b)) en recherchant les meilleures homographies avec un échantillonage local expliqué plus bas. Des ensembles de points appartenant aux mêmes modèles sont ensuite fusionnés (figure 4 (c)) suivant un critère expliqué dans la partie *Fusion des modèles*. Nous retenons enfin les ensembles de points contenant des points 3D incohérents avec la structure statique de la scène (figure 4 (d)).

Notations. Les points 2D détectés dans le plan de la caméra sont représentés par des vecteurs homogènes p. Deux

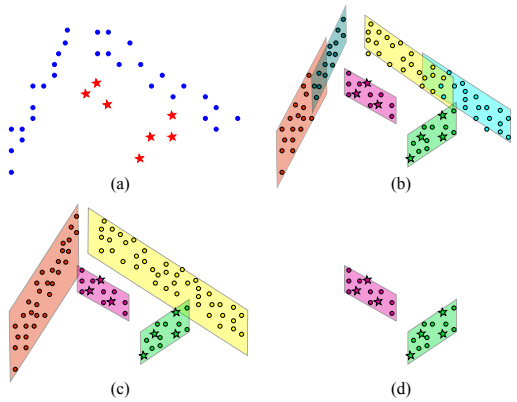


FIGURE 4 – Segmentation des objets. **(a)** Reconstruction 3D des points de la structure statique (points bleus) et points 3D ayant bougé (étoiles rouges). **(b)** Détection des meilleures homographies dans chaque sous-régions de l'image. **(c)** Homographies fusionnées. **(d)** Objets ayant bougé entre les deux passages.

points p_1 et p_2 détectés dans des images similaires sont mis en correspondance. Ce sont les projections du même point 3D. Nous recherchons les surfaces planes de la scène, ce qui revient à trouver les ensembles de points reliés par la même homographie H :

$$p_2 \sim H p_1 \tag{2}$$

La matrice H a 8 degrés de liberté, elle peut être déterminée avec 4 correspondances de points. Dans notre problème, le système est surdéterminé et il faut estimer H en prenant en compte l'ensemble des correspondances vérifiant potentiellement la relation 2. Comme les mesures sont bruitées, la relation 2 n'est pas exactement vérifiée et nous quantifions l'erreur avec la distance de Sampson [7]. La relation peut être écrite sous la forme du système d'équations $Ah = 0$, h étant le vecteur contenant les 9 composantes de la matrice H. La distance de Sampson pour une homographie s'écrit alors :

$$e^2_{Sampson} = h^T A^T \left(J J^T\right)^{-1} A h, \tag{3}$$

où $J = \frac{\partial(Ah)}{\partial(\tilde{p})}$ est la jacobienne du système d'équations, $\tilde{p}$ les coordonnées de deux points appariés. Les correspondances présentant une erreur inférieure à un certain seuil ϵ sont considérées valide pour l'homograhie H (nous prenons $\epsilon = 1,5$). La matrice peut alors être estimée par une méthode de minimisation (moindres carrés) avec l'ensemble des correspondances valides.

Procédure RANSAC séquentielle. Nous souhaitons utiliser une méthode simple et rapide pour calculer les homographies entre deux images similaires. Nous appliquons séquentiellement une procédure RANSAC. Le modèle retenu est le modèle reposant sur le plus grand nombre de points *inliers*. Un premier groupe de points est constitué des correspondances validées par le modèle. On répète ensuite la procédure après avoir retiré les *inliers* du modèle précédant et si le nombre de points restants est suffisant.

Echantillonage local. Les données sont polluées par de nombreuses erreurs (bruit de mesure, erreurs d'appariemment). D'autre part, la scène peut être constituée d'un grand nombre de plans et certains objets peuvent avoir bougés. Au final, chaque modèle d'homographie valide une faible proportion de correspondances et le problème peut rapidement devenir insoluble. Pour pallier ce problème nous générons les modèles candidats en utilisant un échantillonage local comme le font Schindler et al. [13]. Les points d'un même objet sont regroupés dans l'image. En utilisant un ensemble de points d'une même sous-région de l'image nous réduisons le nombre d'itérations de la procédure RANSAC. Les sous-régions utilisées sont présentées figure 5. Les échantillons sont formés par des tirages dans l'image entière, chaque ligne, chaque colonne et chaque région définie par l'intersection d'une ligne et d'une colonne. Nous considérons que les plus petits objets couvrent au moins 50% d'une des sous-régions.

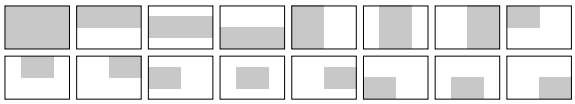


FIGURE 5 – Sous-régions de l'image

Fusion de modèles. Notre algorithme génère des groupes de points $X_1, ..., X_n$ appartenant à des objets plans de la scène. Comme la plupart des hypothèses de départ sont générées à partir de groupes de points spatialement proches, deux groupes distincts peuvent contenir des points globalement sur le même plan. Ces deux groupes doivent être fusionnés. Nous regroupons dans un premier temps les groupes possédant plus de 80% points en commun. Ensuite, pour chaque paire de groupes de points X_1 et X_2 nous estimons une homographie avec l'union des deux ensembles avec la méthode des moindres carrés. Les deux groupes sont fusionnés si la moyenne de l'erreur pour ce nouveau modèle est inférieur au seuil ϵ :

$$\frac{1}{|X_1 \cup X_2|} \sum_{c \in X_1 \cup X_2} e_{\hat{H}}(c) < \epsilon \tag{4}$$

Détection des objets. La première étape de notre algorithme permet de détecter des ensembles de points appartenant à des surfaces planes de la scène (figure 4 (c)). On définit un objet comme une surface plane ayant bougé. L'algorithme de détection de plans est donc fusionné avec la reconstruction 3D pour détecter les objets : nous retenons les ensembles de points contenant des points 3D incohérents avec la structure statique de la scène (représentés par des étoiles figure 4 (d)). Ces ensembles sont enregistrés en tant qu'objets pouvant être détectés par la suite si suffisamment de points ont été reconnus. On observe quelquefois de fausses détections. En effet, si une grande partie de la scène

a bougé, la structure statique est plus difficile à repérer. Certaines zones immobiles sont alors considérées comme ayant bougé. Nous utilisons la redondance des images clefs pour filtrer les fausses détections : un objet doit être détecté plus de trois fois dans la comparaison de deux passages dans un même lieu pour être finalement enregistré.

5 Résultats expérimentaux et discussions

5.1 Validation expérimentale

Nous validons notre système sur deux séquences vidéo. La première séquence est une séquence réelle de 2035 images prises à l'intérieur d'un bâtiment. La figure 7 présente nos résultats de détection d'objets sur quatre cas de reconnaissance de lieu. Les vues 1 et 2 sont deux images provenant du même lieu. On observe les points de la carte 3D reprojetés sur la vue 1 (vue courante) : les points considérés comme appartenant à la structure statique sont en bleu et les points incohérents en rouge. La vue 3 montre les ensembles de points ayant eu un même mouvement entre les deux passages. Les objets ayant effectivement bougés et les ensembles de points correspondant à un objet détecté sont représentés sur la vue 4. Sur les cas 7(a), 7(b) et 7(c), nous détectons correctement le seul objet déplacé. Le personnage apparaissant sur le cas (c) ne perturbe pas les résultats : les points détectés ne sont pas appariés dans l'image similaire. Sur l'exemple (d), trois objets ont été déplacés. Deux objets sont détectés par notre algorithme car les deux livres de gauche ont eu un mouvement très proche. Les objets apparus ou disparus entre les deux passages ne sont pas détecté par notre algorithme.
La deuxième séquence reconstruit en 3D un objet dont on a modifié la géométrie au cours du temps (ouverture du capot et d'une porte sur une maquette de véhicule). Les résultats sont présentés figure 9. L'objet articulé est constitué de plusieurs sous-parties rigides dont nous détectons le déplacement. Nous souhaitons par la suite décrire chaque sous-partie de l'objet séparément.
On peut noter qu'il y a beaucoup de distorsion dans les images car nous utilisons une caméra grand champ mais cela ne pose pas de problème pour le slam métrique car les paramètres internes de la caméra sont connus.

5.2 Détection des objets : améliorations possibles

La figure 8 présentent l'ensemble de nos résultats pour la reconstruction 3D, la reconnaissance de lieu et la détection d'objets. Tous les objets déplacés (six objets) ont été détectés. Nous observons néanmoins deux problèmes récurrents : la présence d'*outliers* dans certains objets et la détection d'ensemble de points n'ayant effectivement pas bougé.

Gestion des *outliers*. Certains points sont inclus par erreur dans l'ensemble des points d'un objet (en rouge figure 7(d)). Ces erreurs sont filtrées comme suit : on estime la moyenne et la variance des coordonnées spatiales et on rejette les points en dehors de l'intersection $[\bar{x} - 2\sigma_x, \bar{x} + 2\sigma_x] \cap [\bar{y} - 2\sigma_y, \bar{y} + 2\sigma_y]$. σ_x et σ_y sont les variances des abscisses et des ordonnées évaluées avec l'estimateur MAD (*Median Absolute Deviation*). $\bar{x}$ et $\bar{y}$ sont les médianes des abscisses et des ordonnées.

Fausses détections. Notre algorithme détecte des ensembles de points associés à des objets réels. Certains ensembles sont détectés par erreur (faux positifs) notamment lorsqu'une grande partie de la scène a bougé. Nous comptons pour la séquence étudiée 18 faux positifs sur 70 détections. Jusqu'ici, nous filtrons ces erreurs en ne gardant que les objets détectés plusieurs fois. Nous pourrions diminuer le taux de faux positifs en utilisant des *a priori* provenant de l'analyse de l'image précédente si elle permet d'établir de façon précise la structure statique de la scène.

FIGURE 6 – Mise en correspondance des points entre deux images similaires : les points appariés en bleu, les points non appariés en vert sur la vue gauche.

Apparition et disparition d'objet. Nous détectons pour l'instant les objets déplacés dans un même lieu : le point de vue de la caméra a légèrement changé et les arrière-plans de l'image courante et de l'image de la base de données sont similaires. Nous souhaitons à l'avenir détecter un objet qui disparait et réapparait dans un lieu différent. La figure 6 présente un ensemble de points non appariés (en vert) entre deux images similaires. Si l'objet réapparait dans un autre lieu, l'analyse des points et de leur mouvement permet de définir de la même façon un objet.

Objets 3D. Nous avons simplifié le problème en recherchant des objets plans. Nous souhaitons aussi détecter des objets 3D. Nous devons pour cela utiliser des algorithmes de sélection de modèles pour déterminer les meilleures homographies et les meilleures matrices fondamentales entre deux images similaires.

6 Conclusion et perspectives

Nous avons présenté une méthode pour détecter de façon automatique des objets. Les différentes explorations de la caméra dans le même environnement permettent de modéliser les objets détectés. Les expériences présentées ont permis de mettre en évidence les performances de la méthode dans un cadre réel de localisation en environnement intérieur avec apprentissage d'objets ayant été déplacés. Nous souhaitons dans le cadre de la généralisation de ces tra-

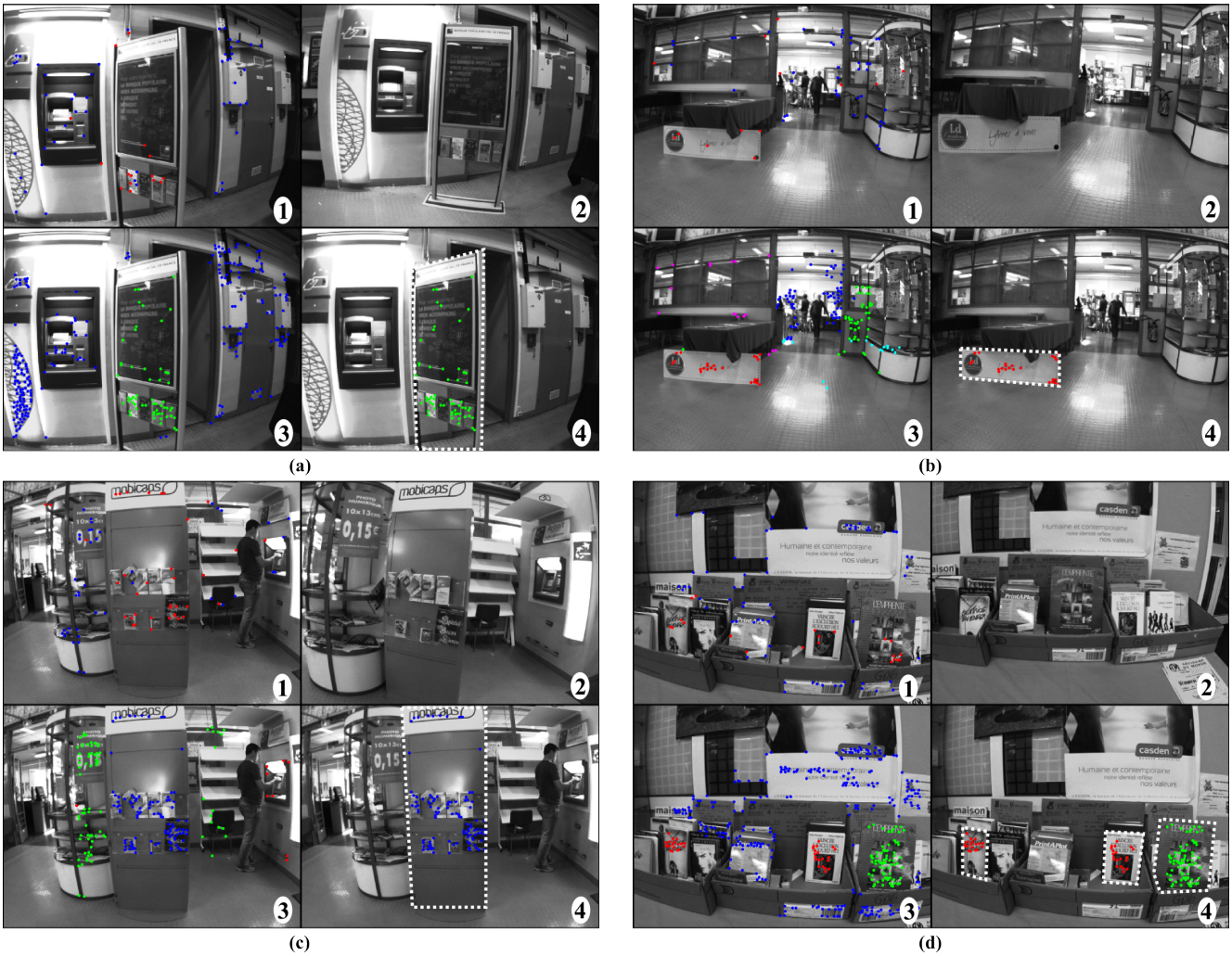


FIGURE 7 – Résultats de la détection d'objets sur quatre cas de reconnaissance de lieux. Vue 1 : image courante. Vue 2 : image similaire extraite de la base de données, en bleu les points de la structure statique, en rouge les points incohérents. Vue 3 : différentes homographies calculées entre les deux vues. Vue 4 : En pointillés blancs les contours des objets effectivement déplacés, en couleur les ensemble de points correspondant à un objet détecté par notre algorithme.

vaux modéliser un environnement intérieur sujet à de nombreuses modifications, comme des ateliers d'usine ou des centres commerciaux, et maintenir une carte des lieux et des objets fréquemment observés. Nous voyons de nombreuses applications à ces travaux. Dans le cadre de scénarios mettant en scène une intéraction de l'utilisateur avec son milieu, il est primordial de comprendre le contexte dans lequel évolue la caméra et d'identifier les objets présents dans l'environnement : la détection et la reconnaissance des objets permet ainsi une meilleure compréhension de la scène. Nous souhaitons aussi améliorer les résultats de la reconnaissance de lieu et de la localisation métrique en gérant les hypothèses de scène non statique. La connaissance d'objets de taille connue peut par la suite être utilisée pour corriger la dérive du facteur d'échelle. Enfin, la détection d'objets et de mouvements dans la scène peut être utile à des applications de réalité augmentée.

Références

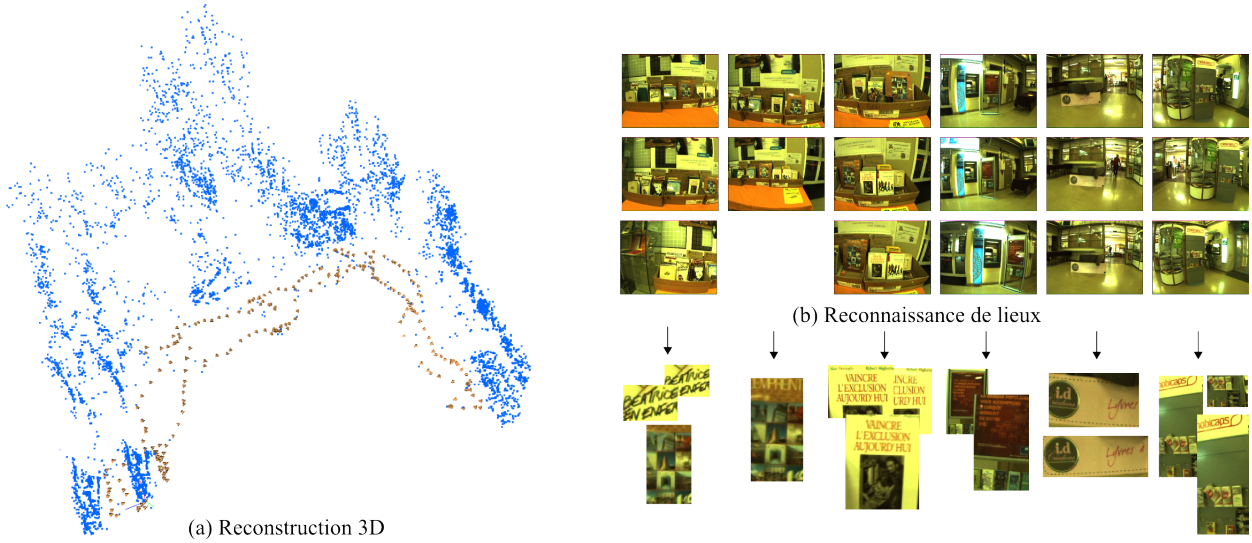
[1] A. Angeli and A. Davison. Live feature clustering in video using appearance and 3D geometry. In *BMVC*, 2010.

[2] A. Angeli, D. Filliat, S. Doncieux, and J.-A. Meyer. A fast and incremental method for loop-closure detection using bags of visual words. *IEEE*, 2008.

[3] H. Bay, T. Tuytelaars, and L. Van Gool. SURF : Speeded Up Robust Features. *ECCV*, 2006.

[4] R. O. Castle, G. Klein, and D. W. Murray. Combining monoslam with object recognition for scene augmentation using a wearable camera. 2008.

[5] R. O. Castle, G. Klein, and D. W. Murray. Video-rate localization in multiple maps for wearable augmented reality. In *SWC*, 2008.

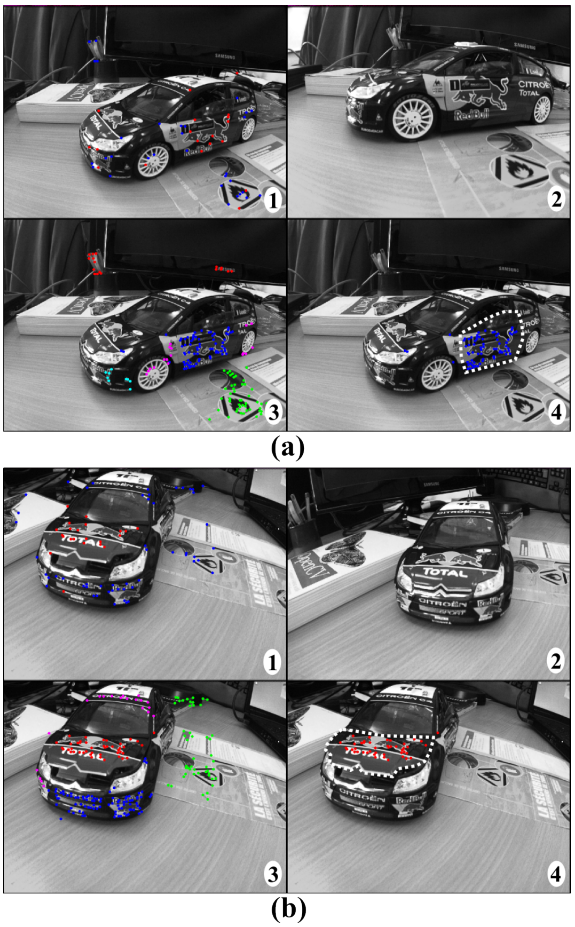


(a) Reconstruction 3D

(b) Reconnaissance de lieux

(c) Objets détectés

FIGURE 8 – **(a)** Reconstruction 3D d'une scène intérieure, **(b)** Images appariées par la reconnaissance de lieu, **(c)** 6 objets détectés dans la scène.



(a)

(b)

FIGURE 9 – Maquette de voiture. Détection de la porte **(a)** et du capot **(b)** ouverts lors du deuxième passage.

[6] M Cummins and P. Newman. Highly scalable appearance-only SLAM : Fab-map 2.0. In *RSS*, 2009.

[7] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. 2000.

[8] K. Kim, V. Lepetit, and W. Woo. Keyframe-based Modeling and Tracking of Multiple 3D Objects. *ISMAR*, 2010.

[9] G. Klein and D. Murray. Parallel tracking and mapping for small AR workspaces. In *ISMAR*, 2007.

[10] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd. Monocular Vision Based SLAM for Mobile Robots. *ICPR'06*.

[11] D. Nister and H. Stewenius. Scalable Recognition with a Vocabulary Tree. *CVPR'06*.

[12] G. Reitmayr, E. Eade, and T. Drummond. Semi-automatic annotations in unknown environments. In *Proc. ISMAR 2007*.

[13] K. Schindler and S. Suter. Two-view multibody structure-and-motion with outliers through model selection. *PAMI*, 2006.

[14] J. Sivic and A. Zisserman. Video Google : a text retrieval approach to object matching in videos. *ICCV*, 2003.

[15] R. Toldo and A. Fusiello. Robust multiple structures estimation with j-linkage. In *ECCV*, 2008.

[16] P.H.S. Torr. Model selection for two view geometry : A review. *Microsoft Research*, 1998.

[17] M. Zuliani, C. S. Kenney, and B. S. Manjunath. The multiransac algorithm and its application to detect planar homographies. In *ICIP (3)*, 2005.